National Security Agency

Information Assurance Directorate

Vulnerability Analysis and Operations

Systems and Network Analysis Center

Application Whitelisting using Software Restriction Policies

Contents

Figures

Tables

Abstract

Software Restriction Policies (SRP) enables administrators to control which applications are allowed to run on Microsoft Windows. SRP is a feature of Windows XP and later operating systems. It can be configured as a local computer policy or as domain policy using Group Policy with Windows Server 2003 domains and later. Using this guide, administrators can configure SRP to prevent all applications in their domain from running except applications they explicitly allow. Utilizing SRP as an application whitelisting technique significantly increases the security posture of the domain by preventing some malicious programs from executing.

For some additional government-only implementation details, please read the addendum to this document at https://www.iad.gov/library/snac.cfm (DoD PKI required), at http://www.iad.smil.mil/resources/library/ on SIPRNet, by request from your IAD Customer Advocate, or by contacting the NSA IA Service Center at NIASC@nsa.gov or the NSA SNAC at snac@radium.ncsc.mil.

Introduction

The amount of malware on the Internet increases in volume and variety every day. Malware developers and antivirus vendors are in a never-ending arms race. Malware authors continuously modify their creations so they are not detected, and antivirus vendors update their software daily to detect new malware variants. Defending against these threats by blocking every known malware sample, a technique known as *blacklisting*, is a reactive technique that does not scale well to the increasing volume and variety of malware. It also does not protect against unknown malware. Many attacks use previously unknown vulnerabilities, also known as zero-day vulnerabilities, which cannot be prevented with blacklisting techniques.

Corporate and government networks are prime targets for attackers. They contain valuable proprietary or sensitive information and have a large, diverse attack surface for an adversary to exploit. Attacks have shifted from operating system attacks to application-based attacks. This change has left each individual user, and the applications they use, as the main attack vectors into the network.

Application *whitelisting* is a proactive technique where a limited number of programs are allowed to run, while all other programs are blocked from running by default. Below are some example scenarios that may be mitigated by using whitelisting:

- A user runs a program that looks like a greeting card or a streaming video viewer that executes a hidden malicious program.
- An attacker exploits a vulnerable program and downloads a malicious program to further compromise the network and steal data.
- A user views a web site that silently exploits a previously unknown or unpatched vulnerability in their browser, or third-party browser add-on, and then executes a malicious program to steal the user's data.
- A user opens a document that exploits a vulnerability in the document viewer and a malicious program is unknowingly executed.
- A user inserts removable media into their computer, such as a USB thumb drive, that automatically executes a malicious program.
- A user installs a program not allowed by policy.

Since none of the malicious programs in the above scenarios are included in the list of allowed programs, they should not be executed. Whitelisting makes it more difficult for attackers to compromise a network because they must exploit one of the allowed programs on the victim's computer or circumvent the whitelisting mechanism to perform a successful attack. Even if an allowed program is exploited, further malicious activity may still be blocked by the whitelisting mechanism.

Application whitelisting is not a replacement for traditional security software. It should be used as one layer in a defense-in-depth solution. For an application whitelisting solution to be effective:

- All executable code must be blocked by default so only approved programs can run.
- Users must not be allowed to run programs from directories where they can save files.
- Users must not have administrator privileges.

Microsoft Windows operating systems include a feature called Software Restriction Policies (SRP). Administrators can configure SRP as an application whitelisting solution where only specific executables are allowed to run while all other executables are prevented from running. SRP can also limit which application

libraries may be loaded by executables. SRP is feature of Windows XP and later operating systems. It can be configured as a local computer policy or as domain policy using Group Policy with Windows Server 2003 domains and later.

About this Guide

This guide describes SRP settings recommended by the NSA Information Assurance Directorate's (IAD) Systems and Network Analysis Center (SNAC) and provides administrators with a walkthrough for implementing the settings.

Using SRP as an application whitelisting solution will not stop all malicious software. It provides an additional layer in a defense-in-depth strategy. The intent of this guidance is to prevent users from unknowingly or accidentally executing malicious code.

Please read this document before implementing the guidance. Any configuration changes should be validated on a test network or on a small set of test computers to ensure the settings are correct before making changes to the entire domain.

There are many references that describe how to configure SRP. This document is not meant to replace those resources and does not explain all possible configuration options. For more information about SRP, consult the Microsoft documents "Using Software Restriction Policies to Protect Against Unauthorized Software"[1] and "Windows Server 2003 Technical Library: Software Restriction Policies."[2]

For some additional government-only implementation details, please read the addendum to this document at https://www.iad.gov/library/snac.cfm (DoD PKI required), at http://www.iad.smil.mil/resources/library/ on SIPRNet, by request from your IAD Customer Advocate, or by contacting the NSA IA Service Center at NIASC@nsa.gov or the NSA SNAC at snac@radium.ncsc.mil.

Related Tools

A number of sample scripts have been developed by the NSA SNAC to help deploy SRP and give administrators some flexibility in customizing SRP settings for their unique environment. For more information about these scripts, see Appendix A, Appendix C, and Appendix D. These scripts are distributed with this guide.

Known Issues

There is a known bug in the SRP implementation for certain versions of Windows. A hotfix from Microsoft is available to fix this bug and should be applied to all affected computers before enabling SRP. Some minor usability issues may occur when using SRP, especially if the hotfix is not applied, that could annoy users, such as double-clicking documents on network shares not launching their associated document viewer application. Software update mechanisms that require users to apply patches to programs may no longer function once SRP whitelisting is enforced, but automatic update mechanisms should not be affected. Due to these issues, SRP settings should be thoroughly tested on a limited set of computers that have all deployed software before being applied in a production environment. See Appendix D for more information about these issues.

---

[1] "Using Software Restriction Policies to Protect Against Unauthorized Software" http://technet.microsoft.com/en-us/library/bb457006.aspx

[2] "Windows Server 2003 Technical Library: Software Restriction Policies" http://technet.microsoft.com/en-us/library/cc779607.aspx

Implementation

The following steps state how to implement SRP in an Active Directory domain and are described in further detail below. Implement SRP by following these steps:

1. Audit the domain to determine which applications are running on domain computers.
2. Configure SRP to run in whitelisting mode.
3. Decide which applications should be allowed to run and create additional SRP rules as needed.
4. Test the SRP rules and modify or create additional rules as necessary.
5. Deploy SRP to successively larger Organizational Units until SRP is applied to the whole domain.
6. Monitor SRP on an ongoing basis and modify the rules when appropriate.

Audit the Domain to Determine which Applications are Running

Before applying SRP, it is important to understand which applications are running on domain computers. An audit of the domain is essential for creating a set of robust SRP rules that will enable users to continue running authorized programs that are stored in non-default locations. Programs running from the paths specified by the SYSTEMROOT and PROGRAMFILES environment variables, usually C:\Windows\ and C:\Program Files\, can usually be ignored during an audit except if those programs load libraries from other paths or are writeable by regular users. Those paths are not writable by regular users in default installations of Windows and they are the built-in allowed paths for the default SRP settings. Tools and scripts exist that can be used to create a list of currently running executables on computers for use in a domain audit.

Configure SRP to Run in Whitelisting Mode

The following section explains how to configure SRP in an Active Directory environment to run in application whitelisting mode with the most secure settings. The screenshots are for Windows Server 2003, but differences for Windows Server 2008 have been noted in the text. To apply SRP to the domain:

1. Create a new Group Policy Object (GPO). Give the GPO a name that can be easily associated with SRP.
2. Open the newly created GPO for editing in the Group Policy Object Editor in Windows Server 2003 or the Group Policy Management Editor in Windows Server 2008.
3. Go to **User Configuration** → **Windows Settings** → **Security Settings** → **Software Restriction Policies** as shown in Figure 1. When configuring SRP for the first time, the message shown on the right side of Figure 1 will be displayed. Note that in Windows Server 2008, the **Policies** node exists between the **User Configuration** and **Windows Settings** nodes.

   By creating the policy as a user policy rather than a computer policy, the SRP settings can be applied to specific users or groups so that whitelists can be enforced at a more granular level. For example, accounts with domain administrator privileges can be put in an Organizational Unit (OU) that does not have the SRP settings applied. This allows domain administrators to be exempted from the SRP settings, while forcing local computer administrators to be affected by them. Another advantage is that a general whitelist can be applied domain wide and then more specialized whitelists can be applied to specific OUs depending on the needs of different parts of an organization.

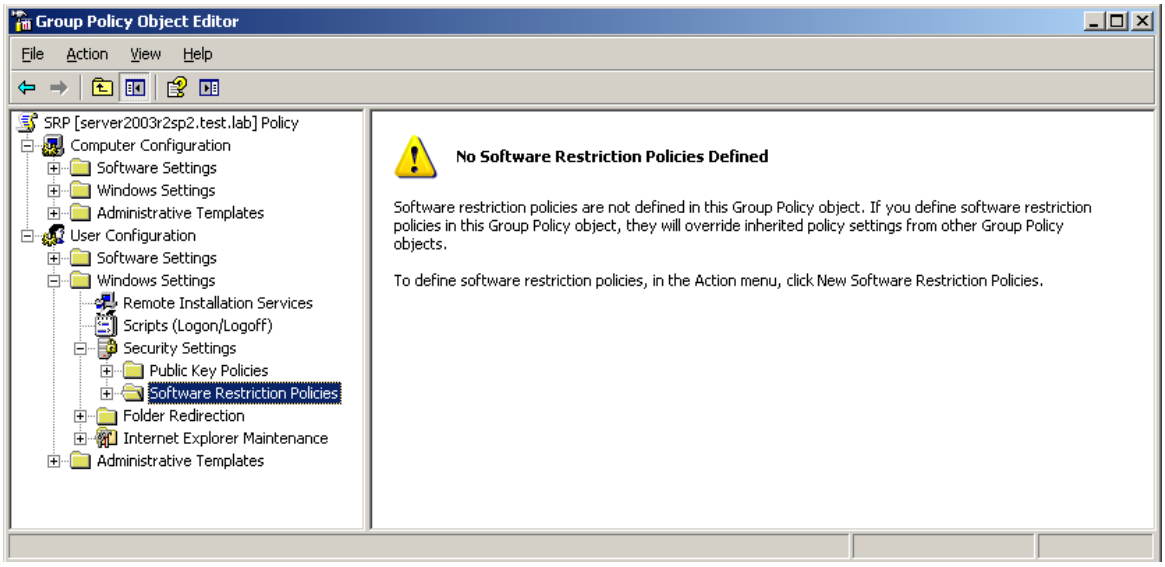

Figure 1: Select the Software Restriction Policies object in the Group Policy Object Editor.

4. Right-click on the **Software Restriction Policies** policy object and select **New Software Restriction Policies** from the menu as shown in Figure 2.

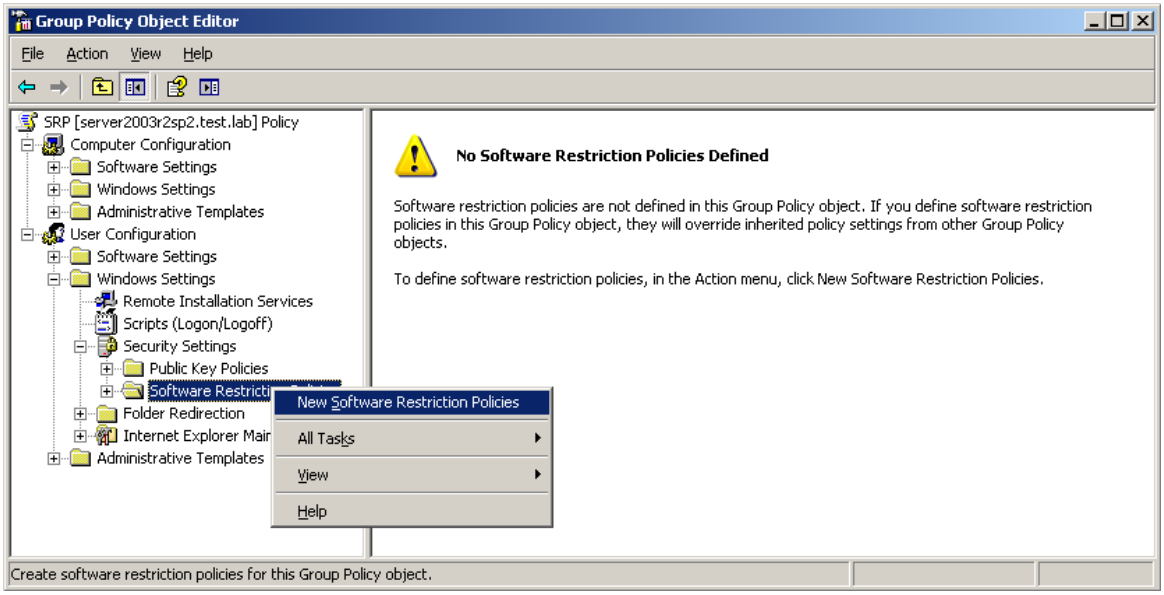

Figure 2: Select New Software Restriction Policies from the right-click menu.

5. Once the new policy is created, select the **Enforcement** policy object that is listed in the right-hand pane, as shown in Figure 3, when the **Software Restriction Policies** policy object is selected.

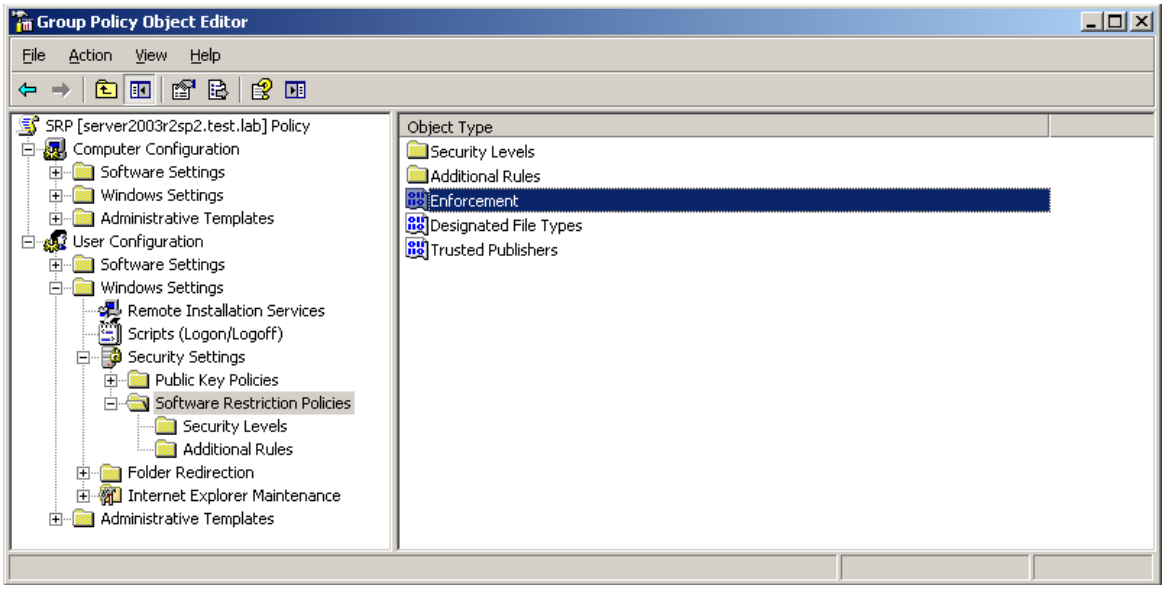

Figure 3: Select the Enforcement policy object.

6. Double-click the **Enforcement** policy object to open the **Enforcement Properties** dialog as shown in Figure 4. Select the **All software files** radio button so SRP will be applied to both executables and libraries. Select the **All Users** radio button so SRP will be applied to all domain users including local administrators. Click the **OK** button when finished.

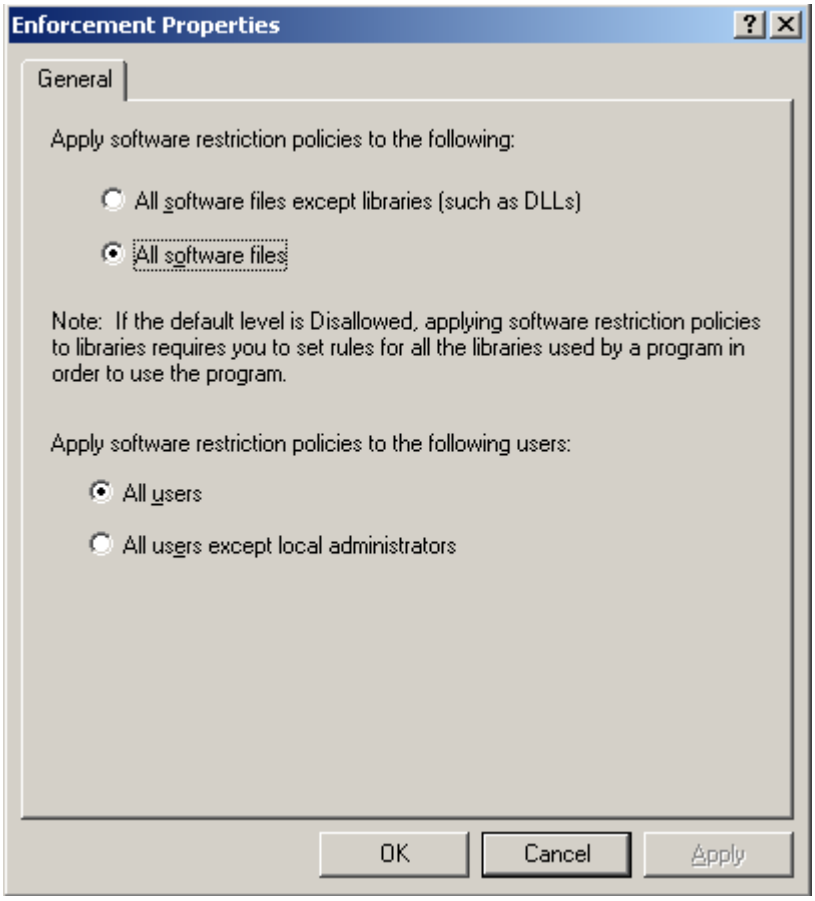

Figure 4: Select the most secure options in the Enforcement Properties dialog.

7. In the Group Policy Object Editor, click on the **Security Levels** policy object to configure the mode SRP will operate in. SRP can operate in blacklist mode or whitelist mode. Blacklist mode is where *all* applications are allowed to run except the ones an administrator specifically denies. Whitelist mode is where *no* applications are allowed to run except the ones an administrator specifically allows. Configuring SRP to use whitelist mode is the most secure and recommended mode. Double-click the **Disallowed** security level and then click the **Set as Default** radio button as shown in Figure 5.

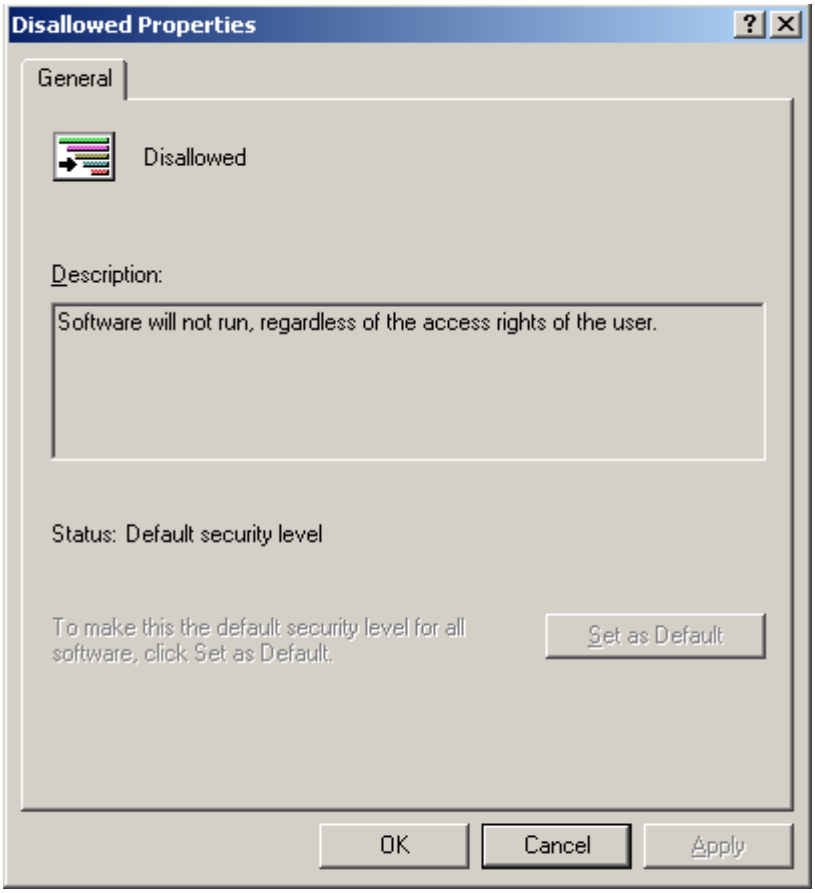

Figure 5: Set Disallowed as the default security level.

After clicking the **Set as Default** radio button, a dialog may appear with the warning:
The default level you selected is more restrictive than the current default security level. Changing to this default security level may cause some programs to stop working. Do you want to continue? Click the **Yes** button.

After completing the above steps, all programs are not allowed to run except for ones in paths specified by the SYSTEMROOT and PROGRAMFILES environment variables, usually C:\Windows\ and C:\Program Files\. These path rules are automatically added when the **Disallowed** security level is set as the default. The rules can be viewed by clicking on the **Additional Rules** policy object and are also documented in Table 1.

Server	Default Rules
Windows Server 2003	%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot% %HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%*.exe %HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot%System32*.exe %HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\ProgramFilesDir%
Windows Server 2008	%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SystemRoot% %HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\ProgramFilesDir%

Table 1: Default SRP Path Rules

Create SRP Rules for Authorized Applications

Now that SRP is configured in a whitelisting mode with the most secure settings, new rules can be added based on the results from the domain audit. The results from the domain audit will determine what additional paths may be needed when creating new path rules. SRP supports four rule types, but only path rules are used in this

guide because they are the easiest to administer and have the least impact on system performance. When more software publishers consistently digitally sign all their application files and computers are faster so the performance impact will be less noticeable, then SRP rules should be transitioned to use the more secure digital signature rules.

Path rules use local or universal naming convention (UNC) paths (e.g., \\server\share) of a file or folder. They support the wildcard characters of ***** (match many characters) and **?** (match one character). Path rules also support registry paths. Registry path rules are identified by percent signs that surround the entire path of the registry entry. When the rule is evaluated, the value of the specified registry entry is used. Do not confuse registry path rules with regular path rules that use environment variables. Environment variables are also surrounded by percent signs but only for the variable rather than the entire path. Some common paths to consider for additional path rules are listed in Table 2.

Description	Type	Rule
The path of the Program Files (x86) folder on 64-bit computers.	Registry Path	%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\ProgramFilesDir (x86)%
Domain login scripts.	Path	\\%USERDNSDOMAIN%\Sysvol\
Shortcuts such as those used on the Start menu.	Path	*.lnk
CD and DVD drives.	Registry Path	See Appendix A.
Shared domain folders.	Path	See Appendix B.
User programs.	Registry Path	See Appendix C.

Table 2: Common Paths to Consider for New Path Rules

Since libraries loaded by programs are also checked, adding a path rule for the program alone may not be enough to allow the program to execute. Rules may need to be added for any libraries that are not loaded from an allowed path. **Before adding a new path rule, make sure the path is not writable by regular users**. If regular users can write to the location specified in the new path rule, then they can easily bypass the intended SRP policy and run any program they want.

To add a new path rule:

1. Right-click on the **Additional Rules** policy object as shown in Figure 6.
2. Select **New Path Rule** from the right-click menu.

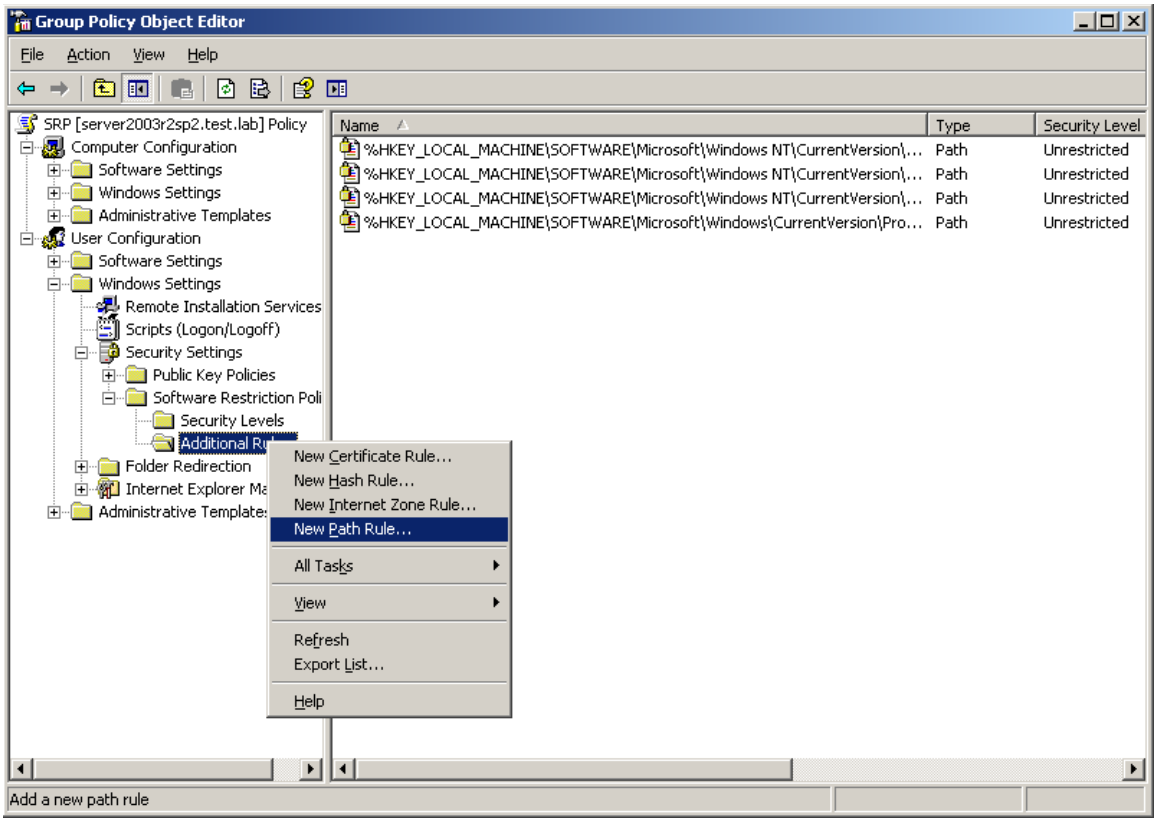

Figure 6: Select New Path Rule from the Additional Rules right-click menu.

3. At the **New Path Rule** dialog, enter a path in the **Path** textbox or click the **Browse** button to select a path as shown in Figure 7.
4. Make sure the **Security level** dropdown menu has the **Unrestricted** option selected.
5. Enter a description in the **Description** textbox if desired.
6. Click the **OK** button to close the **New Path Rule** dialog.

Figure 7 is an example of adding a new path rule for the path of C:\UserPrograms\. By adding this rule, all programs in that path will be allowed to run as long as all the libraries loaded by the programs are loaded from this path or other allowed paths.

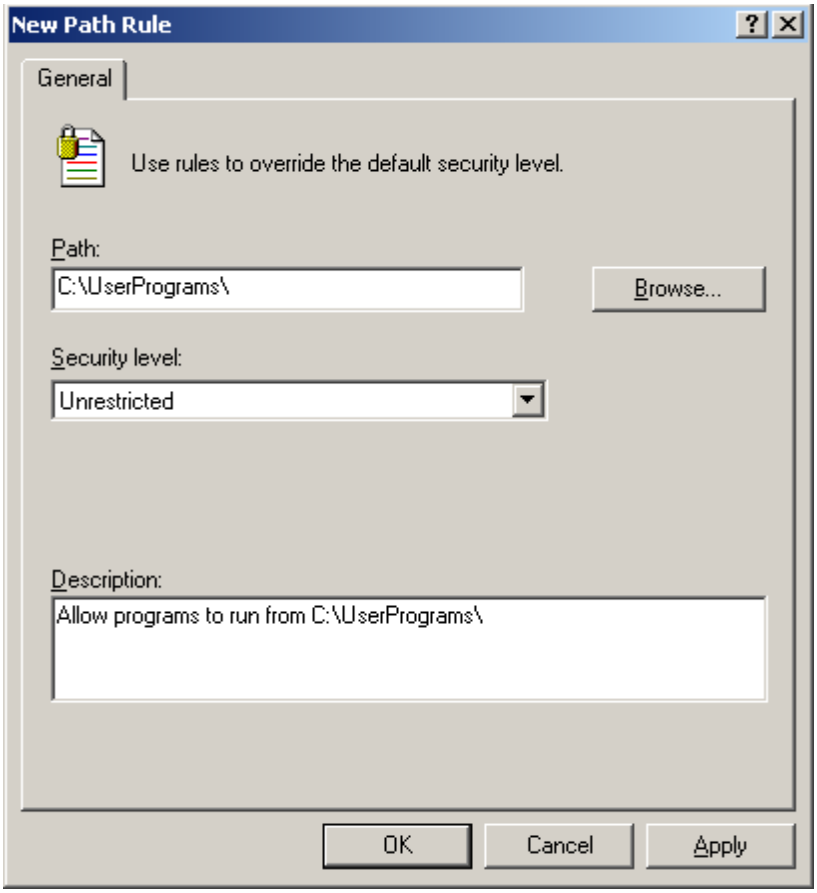

Figure 7: Add a rule for C:\UserPrograms\ in the New Path Rule dialog.

In addition to adding rules for allowed applications that are not in the default allowed paths, administrators should also examine the results of the domain audit to see if there are any unauthorized applications that have been installed within allowed paths and their subfolders. If an unauthorized application is discovered, then uninstall the application. If the application cannot be uninstalled, then create a blacklist SRP path rule specifically for the path of the unauthorized application's subfolder. This blacklist rule can be created by selecting the **Disallowed** option from the **Security level** dropdown menu when creating a rule as discussed in step 4 above. This strategy can make SRP rule management more complex.

Test SRP

After applying SRP, it is important to test that the settings are being applied as intended. Group policy settings are refreshed approximately every 90 minutes, so new settings will not be enforced immediately. Before testing SRP, run the gpupdate command and log off the computer so the new policy will be applied without waiting for the next group policy refresh. If the Fast Logon group policy setting is enabled, then the client computer may need to be rebooted for the new policy to be completely applied. If this behavior is not desired, then the Fast Logon option can be turned off by going to **Computer Configuration** → **Administrative Templates** → **System** → **Logon** and setting **Always wait for the network at computer startup and logon** to **Enabled**.

Test Allowed and Disallowed Paths

Now that the settings are applied as intended, try executing programs from paths that should and should not be allowed by the SRP rule set. When running a program from a disallowed path, a message box should be displayed with the following message on Windows XP and Windows Server 2003: *Windows cannot open this*

program because it has been prevented by a software restriction policy. For more information open Event Viewer or contact your system administrator.

When running a program from a disallowed path at the command prompt, the following message should appear on Windows XP and Windows Server 2003: *The system cannot execute the specified program*.

Windows Vista, Windows Server 2008, and later operating systems use the same error message regardless of how the program is executed: *This program is blocked by group policy. For more information, contact your system administrator*.

If one of the above messages appears for a program that is running from an allowed path, then the program may be loading libraries from a disallowed path. Additional rules may need to be added for the libraries to allow the program to execute. The results from the domain audit will have the paths of the libraries loaded by a program.

Troubleshoot Rules

SRP has some logging abilities that can help when testing or troubleshooting SRP rules. When SRP blocks a program from executing, a Windows Event Log entry should appear in the Application log. Table 3 shows the different Windows Event Log entries related to SRP and their meanings.

Event	Message	Meaning
865	Access to %program% has been restricted by your Administrator by the default software restriction policy level.	A program was prevented from executing due to the default rule which automatically blocks all programs unless they are specifically allowed.
866	Access to %program% has been restricted by your Administrator by location with policy rule %guid% placed on path %path%.	A program was prevented from executing due to a configured path rule.
867	Access to %program% has been restricted by your Administrator by software publisher policy.	A program was prevented from executing due to a SRP certificate rule from a software publisher's certificate.
868	Access to %program% has been restricted by your Administrator by policy rule %guid%.	A program was prevented from executing due to a SRP hash or zone rule.
882	Access to %program% has been restricted by your Administrator by policy rule %guid%.	A program was prevented from executing by SRP but the SRP notification dialog was blocked from showing.

Table 3: Windows Event Log Entries for SRP

When implementing this guidance, event ID 865 in the Application log will be the most common event. The event's description will list the path of the program that was prevented from running. If the path of the program appears to be an allowed path, then it may have been prevented from running due to loading a library from a path that does not have an allow rule associated with it. Consult the domain audit to see what libraries are loaded by the program in question and create new path rules as needed.

Microsoft Sysinternals Process Monitor may also help discover what libraries may have prevented the program from running. To troubleshoot with Process Monitor, run it on a computer that has SRP disabled and then follow these steps:

1. Start Process Monitor and create a new filter.
2. In the **Process Monitor Filter** dialog, configure the drop down menu options so it reads as **Process Name is *program.exe* then Include**.
3. Click the **Add** button and then click the **OK** button.
4. Once the capture has started, ensure that only the **Show File System Activity** option is enabled.
5. Run the program.

6. Look under the **Path** column in the output for any DLLs that may not be in an allowed path.
7. Create new path rules as needed.

If the program still is being prevented from running, then check the Application log for event ID 866. This event will be logged due to a specific disallowed path rule as discussed at the end of the Create SRP Rules for Authorized Applications section. Check the current SRP policy for any specific disallowed path rules that may apply to the path listed in the event's description.

In addition to Windows Event Log entries, a computer specific SRP log file can be created. The SRP log file records the specific rule used by SRP when it examined a program or library to determine if the program should be allowed to run. Create a new registry string value (REG_SZ) named **LogFileName** under the registry key of **HKLM\Software\Policies\Microsoft\Windows\safer\codeidentifiers**. The registry value's data can be set to a file path such as C:\srp.log. New log entries are appended to the end of the file. An example entry from the log file looks like:

cmd.exe (PID = 1022) identified C:\Windows\system32\cscript.exe as Unrestricted using path rule, Guid = {191cd7fa-f240-4a17-8986-94d480a6c8ca}

The Globally Unique Identifier (GUID) from the log file maps to the rules stored in the registry under the registry key **HKCU\Software\Policies\Microsoft\Windows\Safer\CodeIdentifiers\262144\Paths**. It is under the HKCU registry hive since SRP is being configured as user policy in this guide. Below the **Paths** registry key, there is a registry key named **{191cd7fa-f240-4a17-8986-94d480a6c8ca}** that matches the GUID from the log file entry. Under that GUID registry key there is a registry value named **ItemData** with its data set to a folder path or registry path. In the case for the above GUID, SRP evaluated a registry path rule for the registry value that the SYSTEMROOT environment variable gets its value from. This is one of the built-in default SRP rules listed in Table 4.

GUID	Rule	OS
191cd7fa-f240-4a17-8986-94d480a6c8ca	%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\ CurrentVersion\SystemRoot%	Server 2003, Server 2008
d2c34ab2-529a-46b2-b293-fc853fce72ea	%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\ CurrentVersion\ProgramFilesDir%	Server 2003, Server 2008
7272edfb-af9f-4ddf-b65b-e4282f2deefc	%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\ CurrentVersion\SystemRoot%*.exe	Server 2003
8868b733-4b3a-48f8-9136-aa6d05d4fc83	%HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\ CurrentVersion\ SystemRoot%System32*.exe	Server 2003

Table 4: Common GUIDs for Built-in Default SRP Rules

In addition to checking the Windows Event Log and using SRP logging files, see Appendix D for other common issues that may occur and some workarounds.

Deploy SRP throughout the Organizational Unit Hierarchy

Since SRP rules are being configured as user policy within GPOs, unique sets of rules can be attached to individual OUs within the Active Directory organizational structure. The SRP rules should be tailored to each group of users within the OU, tested, and then applied to the OU. This technique should be applied successively up the OU hierarchy until SRP has been deployed successfully to the entire domain.

Monitor SRP

Once SRP has been customized, tested, and applied to the domain, monitor that SRP is working as desired. To accomplish this task, use the domain auditing tool that was used at the beginning of this process. Check the results for any executables that may be running from paths that are not allowed. Examine the Windows Event Log to see which applications are being blocked. If they should be allowed to run, then add new SRP path rules for them. Otherwise, educate users about the official policies regarding use of authorized and unauthorized applications. Enabling SRP logging as described in the previous section may also help monitor SRP operation.

Users will require additional programs. They should request that their administrators install the new programs to paths allowed by SRP. If the program needs to be installed into an alternate location, the administrator should ensure that the path is not writable by regular users and then add a new path rule to the SRP policy.

Appendix A: Allow Execution from CD and DVD Drives

It may be necessary to allow execution from CD and DVD drives. The drive letter for CD and DVD drives may not be the same for all the computers in the domain. A computer startup script, deployed through Group Policy, can be used to customize the SRP settings to allow execution for a specific CD or DVD drive in each computer.

The SRPCDDrive.vbs sample script included with this guide can be pushed out as a machine startup script through Group Policy so that it can be applied to an entire domain, or to an OU that contains the desired computer objects, to allow execution from *one* CD or DVD drive on a computer. To deploy the script as a startup script through Group Policy:

1. Create a new GPO with an appropriate name and open the GPO in the Group Policy Object Editor.
2. To ensure that the computers can receive this policy and run the script, select the name of the policy and click on **Properties** from the **Action** menu.
3. Select the **Security** tab.
4. Click the **Add** button.
5. In the **Select Users, Computers, or Groups** dialog, enter the **Domain Computers** group and click the **OK** button to close the dialog.
6. Select the **Domain Computers** group from the list and choose the **Read** and **Apply Group Policy** permissions as shown in Figure 8.
7. Click the **OK** button to close the **Properties** dialog.

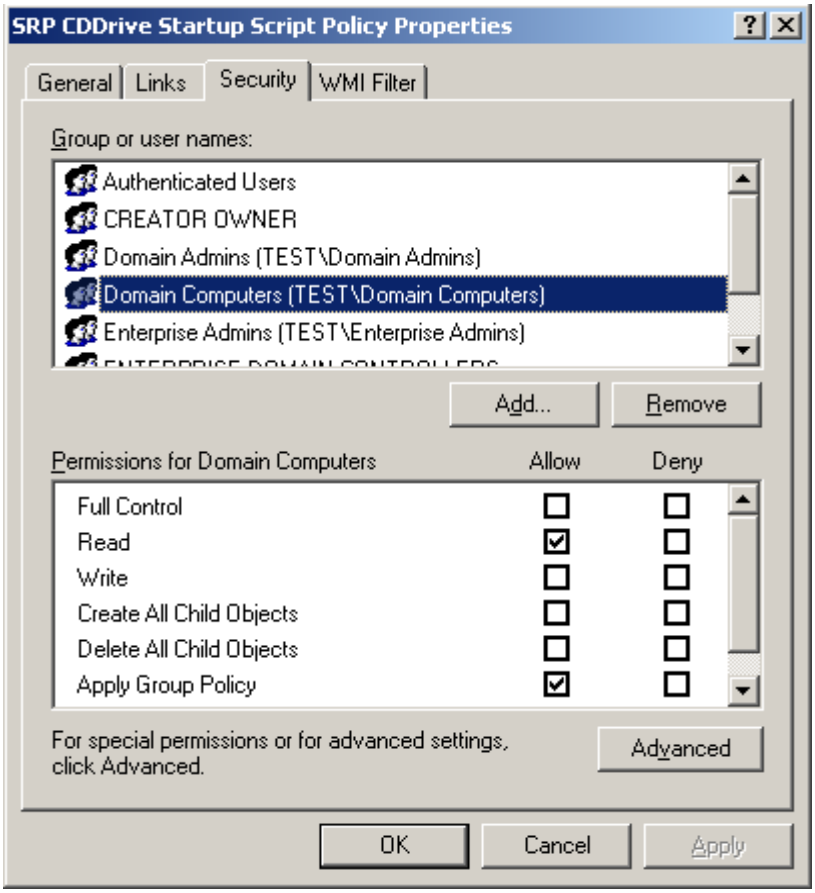

Figure 8: GPO permissions on the Domain Computers group for the startup script.

8. Under **Computer Configuration** → **Windows Settings** → **Scripts (Startup/Shutdown)**, select **Startup**.

9. Right-click on **Startup** and select **Properties** from the menu.
10. At the **Startup Properties** dialog, click the **Add** button.
11. At the **Add a Script** dialog, click the **Browse** button and navigate to the script location.
12. Select the script and click the **Open** button.
13. Click the **OK** button to close the **Add a Script** dialog.
14. Click the **OK** button to close the **Startup Properties** dialog.

The script will create a registry value named **CDROM** under the registry key of **HKLM\Software\Policies\SRP**. The registry value data will be the drive letter of the first CD or DVD drive the script finds on a computer. A Software Restriction Policy registry path rule of **%HKEY_LOCAL_MACHINE\SOFTWARE\Policies\SRP\CDROM%** with a **Security level** of **Unrestricted** can then be created to allow execution from the CD or DVD drive.

Appendix B: Allow Execution from Shared Folders

It may be necessary to allow execution from shared folders in a domain. If there are programs that are available from shared folders, then these programs can be allowed to execute by adding a path rule to the SRP policy with a **Security level** of **Unrestricted**.

For example, if a program is installed on a server at \\SERVER\ProgramDirectory, then that is the path that should be specified in the SRP path rule since SRP path rules support UNC paths. To ensure that no additional programs are allowed to run, the permissions on that directory and its subdirectories should be configured to only allow administrators to write files in that shared folder. If normal users can write to any directory within the path used in a new rule, then they will be able to run any program they want.

Appendix C: Allow Execution from a User Writable Path

It may be necessary in some situations to allow execution from a user writable path so users can install their own programs. This is not recommended because it allows users to run any program they want. If this scenario must be allowed, then it is best to have a unique location for users so that an exploit will not easily be able to copy a malicious program into that location and allow its execution.

The SRPUserPrograms.vbs sample script included with this guide can be used to create a unique location per computer to allow users to install their own programs. The script can be deployed through Group Policy in the same manner as the script in Appendix A.

The script creates a folder at **C:\UserPrograms-####** where the pound signs are a generated number that is unique to the computer. Then the permissions on the folder are changed so only the built-in local Users and Administrators groups are able to write to the folder. The script creates a registry value named **UserPrograms** under the registry key of **HKLM\Software\Policies\SRP** with the path of the folder plus the value of the USERNAME environment variable, the user's login name, as the registry value data.

A new SRP registry path rule of **%HKEY_LOCAL_MACHINE\SOFTWARE\Policies\SRP\UserPrograms%** with a **Security level** of **Unrestricted** can be created to allow execution from the special folder. For example, the script and SRP rule would allow user with a login of "jane" to execute her programs from C:\UserPrograms-1234\jane\. The "jane" folder inside the C:\UserPrograms-1234 folder would need to be created by the user.

Do not use the exact paths and registry keys listed above. They are just examples. Each organization should have their own unique folder path and registry key name to avoid similar implementations across multiple organizations that malware could leverage. No matter what specific registry key name is selected, make sure the key is stored under the HKLM hive so that regular users will not be able to modify the key and its value. The sample script included with this guide can be easily modified to fit an organization's unique needs.

Appendix D: SRP Known Issues and Workarounds

There are some known usability issues and one bug that may affect organizations that adopt this guidance. Most of the usability issues will only be encountered on systems affected by the bug prior to patch installation.

Issues

The following items are some examples of possible issues, and potential workarounds, that users may experience once SRP rules are applied.

- Using the **Open With** submenu from a disallowed path may not work and an error message may not be displayed. A workaround is to move the target of the **Open With** submenu to an allowed path.
- Using the **Send To** submenu from a disallowed path may not work and an error message may not be displayed. A workaround is to move the target of the **Send To** submenu to an allowed path. Also check if the specific **Send To** submenu item uses a shortcut. If it uses a shortcut then the same workarounds for shortcuts discussed below may apply.
- Double-clicking certain types of documents from a disallowed path may not work and a misleading error message may be displayed or an error message may not be displayed at all. Microsoft Office and Adobe PDF files are examples that can have this behavior. A workaround is to open the program first and then open the document by using the **File** menu. Sometimes opening the document using the right-click **Open With** submenu can fix this problem too.
- Shortcuts that have the **Start in** property set to a disallowed path may not work and an error message may not be displayed. A workaround is to modify the shortcut's **Start in** property so it is empty or set to an allowed path. Also check that the shortcut's **Target** property is in an allowed path.
- A program using a working directory that is a disallowed path may not run correctly. A workaround is to start the program using a shortcut with the shortcut's **Start In** property set to an allowed path.
- A program may prompt the user to install an update. The user follows the update prompt to install the update and it appears to install correctly since there was no visible error message. Depending on how the update mechanism works, it may not have actually installed. Some software update mechanisms write to the Temp folder and that folder will not be allowed to execute programs based on the default built-in SRP rules. If the software update is not allowed to execute, there will be an event in the Windows Event Log under the Application log with an event ID of 865. Since the Temp folder is writeable by a user, it should **not** be allowed by SRP. One example of this scenario is the Adobe Flash updater. Application updates should be automatically distributed without relying on user intervention. If possible, configure a patch management solution to distribute these updates or develop a script to deploy the updates without requiring user intervention.

Bugs

There is a known bug when using SRP with the **All software files** option enabled as recommended in this guide. Microsoft has documented this issue in knowledge base article 959074[3]. The operating system internally generates invalid paths when SRP tries to resolve the locations of libraries loaded by an executable. This problem can occur when running a program from drives other than the system drive (usually C:\). Other drives include other physical hard drives, drive partitions, removable drives, and mapped network drives. The problem occurs when a user is logged on to the following operating systems:

[3] Software Restriction Policy Enforcement set to "All Software Files" causes checks against paths/files that are invalid. http://support.microsoft.com/kb/959074

- Windows Server 2003, including R2, with SP1 or SP2
- Windows Vista, Windows Vista SP1, Windows Vista SP2
- Windows Server 2008 (Windows Server 2008 is based off Vista SP1), Windows Server 2008 SP2

Windows XP, Windows 7, and Windows Server 2008 R2 do not appear to be affected by the bug. Microsoft Help and Support has a patch available at http://support.microsoft.com/kb/969972 for Windows Vista SP1, Windows Vista SP2, Windows Server 2008, and Windows Server 2008 SP2. Installing this patch fixes this bug. The patch will be included in Windows Vista SP3 and Windows Server 2008 SP3 as well. Unfortunately patches are not available for:

- Windows Server 2003, including R2, with SP1 or SP2
- Windows Vista with no service pack installed

For the above operating system versions, the knowledge base article recommends disabling the **All software files** option. This recommendation *drastically* lowers the protection offered by Software Restriction Policies. For Windows Vista with no service pack, upgrade to the latest service pack and install the patches. This leaves Windows Server 2003, including R2, with SP1 or SP2 as the only versions that require a workaround. There are two ways to fix the problem:

1. Create a junction to the Program Files and Windows folders on the other drives.
2. Create empty Program Files and Windows folders on the other drives.

Both methods require that SRP path rules be created for the new paths. The SRPWorkaround.vbs sample script included with this guide uses the second method. The script will create Program Files and Windows folders on all non-system hard drives. Then permissions are copied from the Program Files and Windows folders on the system drive to the new folders. If those folders already exist on the non-system drives, then the script will only copy the permissions. Changing the permissions is important because if a user created the folder, then they would have write privileges which would allow them to run any program from the folder.

Note that the script does not create folders on mapped network drives so opening documents from these drives may not work depending on the program used to open the documents. In this case it is recommended to copy the document to the local hard drive, edit the document, and then copy the document back to the network drive location.

Also note that in the case where the Program Files or Windows folders already exist on the other drives, the script will essentially reset the permissions based on what permissions are applied to those same folders on the system drive. This may cause unintended consequences due to an organization's custom permissions settings. Please review the script, modify it as necessary, and test it before deploying it.

The script creates new SRP rules for the paths if the paths are not already covered by existing rules. These rules are created in the local machine's HKLM registry hive. The local machine rules will be merged with the user level rules defined at the domain level. The rules will not show up in the Local Security Policy interface though. The script can be deployed as a machine startup script through Group Policy as outlined in Appendix A.